



Humanoid Teleoperation using Task-Relevant Haptic Feedback

Firas Abi-Farraj, Bernd Henze, Alexander Werner, Michael Panzirsch,
Christian Ott, Máximo Roa

► To cite this version:

Firas Abi-Farraj, Bernd Henze, Alexander Werner, Michael Panzirsch, Christian Ott, et al.. Humanoid Teleoperation using Task-Relevant Haptic Feedback. IROS'18 - IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS'18, Oct 2018, Madrid, Spain. pp.5010-5017, 10.1109/IROS.2018.8593521 . hal-01853934

HAL Id: hal-01853934

<https://hal.inria.fr/hal-01853934>

Submitted on 6 Aug 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Humanoid Teleoperation using Task-Relevant Haptic Feedback

Firas Abi-Farraj¹, Bernd Henze², Alexander Werner², Michael Panzirsch², Christian Ott², and Máximo A. Roa²

Abstract—Robotic teleoperation is a key technology for a wide variety of fields. Teleoperating a humanoid in particular is essential as it allows the user to act remotely on an interface designed especially for humans, e.g., in a space station, or operating tools and machinery in disaster scenarios. This paper presents a ‘task-relevant’ haptic interface for humanoid teleoperation, which bridges the gap between the task at hand and the balance of the robot. The operator is given command over the humanoid’s hands and is informed through haptic cues about the impact of her/his potential actions on the robot’s stability. Moreover, a null-space autonomous controller acts in the operator’s null-space to provide her/him with a wider workspace and help in the successful execution of the task. The architecture is designed to top an existing compliance controller for a torque-controlled humanoid robot. Experiments on the humanoid robot TORO are reported to demonstrate the feasibility and effectiveness of the approach.

I. INTRODUCTION

Robotic teleoperation is playing a pivotal role for allowing human presence in remote and hazardous environments such as nuclear sites, deep oceans or the outer space. Visual and haptic interfaces are usually exploited to provide the human operator with a sense of presence while giving her/him control over the robotic system. However, controlling a complex system such as a humanoid robot using human inputs alone is a challenging task that requires a lot of training and experience [1]–[3]. Partial autonomy and assisted teleoperation frameworks have been proposed to reduce the mental load on the operator and improve her/his performance [4], [5]. Examples in this sense range from virtual fixtures [6], [7], which are task-dependent and require continuous human input, to more complicated shared control schemes aiming at profiting from the human’s supervisory capabilities in guiding an autonomous system [2], [5].

On the other hand, and while many robotic systems are designed for controlled environments such as manufacturing facilities, the need for a robot capable of interacting with interfaces designed for humans is crucial. Teleoperated humanoids have been exploited as a potential solution in different applications ranging from space manipulation [8] to substituting humans in nuclear environments [9], driving a lift truck [10] or piloting an aircraft [11]. A major challenge in this regard is designing a convenient hardware-software human machine interface (HMI) capable of controlling such

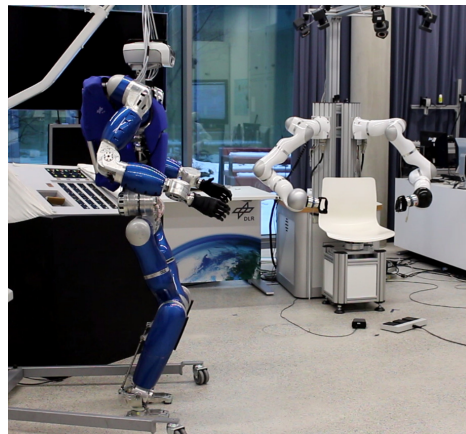


Fig. 1. The Experimental Setup. The haptic user interface system HUG (right) is used as master device, while the torque-controlled humanoid robot TORO (left) works as slave device.

highly versatile robots [12], [13]. The complexity of designing a teleoperation architecture for a humanoid lies in simultaneously giving the user an immersive task-oriented experience while informing her/him at the same time about the robot’s balance and physical constraints. Task-relevant balance feedback is actually a topic that has not been studied thoroughly in literature, and is at the core of the work at hand. Most of the literature either provides the operator with full immersion in the task itself through haptic feedback from interactions with the environment, or gives her/him control over the posture of the humanoid while maintaining its balance. The need for bridging this gap between the robot’s balance and the task at hand is indeed essential.

In [14], the teleoperation system used for controlling the DRC-HUBO humanoid robot at the 2013 DARPA Robotics Challenge trials is presented. In this work, three operators were in charge of fully controlling the humanoid through a software interface for navigation and manipulation purposes. One of the operators was explicitly responsible of ensuring stability and avoiding collisions by commanding the robot through that interface.

A different approach is described in [15], where a single human operator commands a humanoid by acting on a 3 DoF (degrees of freedom) master device. The operator selects a certain point of the robot’s body for manipulation, instead of simultaneously controlling all its DoFs. An autonomous controller integrates the operator’s commands for producing the required whole-body motion while respecting postural stability constraints. In [16], the operator is given control over the robot’s posture via a PHANTOM Omni master device, over which she/he receives haptic cues, reflecting

¹The author is with CNRS at Irisa and Inria Rennes Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes Cedex, France. firas.abi-farraj@irisa.fr

²The authors are with the German Aerospace Center (DLR), Institute of Robotics and Mechatronics, 82234 Wessling, Germany. <firstname>.<lastname>@dlr.de

sensory data from the load cells installed at the robot's feet. The user's actions are logged and used to teach the robot how to keep balance using a learning-from-demonstration framework. A bimanual master interface using two 6 DoF master devices to give the operator command over the humanoid's legs was proposed in [17]. The operator was fed back with force cues informing her/him about the position of the Zero-Moment Point within the support polygon. The same measure was used to give the operator a sense of the robot's balance using a vibrotactile belt for providing cutaneous haptic balance feedback [18]. This cutaneous belt was replaced with a kinesthetic system applying forces on the operator's waist to inform her/him about their proximity to the edges of the support polygon in [19].

The focus of the described literature is either on the manipulation task itself or on the balance/stability of the robot. In the former case, the operator is given command over the DoFs of interest while an autonomous algorithm generates the required whole-body motion and enforces stability constraints. She/he is fed with haptic information related to the task at hand, e.g., interactions with the environment, without any insight on the balance of the robot. In the latter case, the user is given the task of maintaining the robot's balance by acting on a specific HMI to control the lower body (posture, legs, ...) of the humanoid.

The work in [18] is one of few that tackles both aspects simultaneously. Here, the operator is commanding the hands of the humanoid while being informed with haptic cues of its stability through a vibrotactile belt. However, she/he is still not informed about the impact of her/his manipulative actions on stability itself. In fact, manipulation actions have a crucial impact on stabilizing/destabilizing the robot. The operator, who is given command over some of the robot's DoF, can be oblivious as to how these DoF can be employed for improving the robot's balance. To this end, we propose in this paper a balance-feedback human machine interface that closes the gap between the manipulative actions and their impact on the stability of a humanoid. A haptic interface establishes the direct association between the two by providing the operator with cues informing her/him of the impact of one on the other. These cues are meant to provide the user with potential solutions to assure a successful task completion while accounting for the different constraints of the system.

On the other hand, the null space of the operator's commands is also employed to maintain the balance of the robot through an autonomous controller acting on an underlying impedance control architecture previously presented in [20]. In literature, a large variety of balancing controllers can be found, most of them based on inverse dynamics [21]–[23]. Our approach, presented in [20], is passivity-based and allows to stabilize the Center of Mass and the end-effectors that are not used for supporting the robot (the hands, in this context) in Cartesian space with respect to the world. This controller is then suitable as a tool for the teleoperation approach presented in this work.

The rest of the paper is divided as follows: Section II

describes the modeling of the humanoid and the employed compliance controller, section III gives an account on the shared-control architecture commanding the slave's motion and the haptic interface, and the experiments and results are discussed in section IV. Finally, section V concludes the paper and discusses potential future work.

II. BACKGROUND

A. Dynamic Model

In legged humanoid robotics the use of dynamic models with a free-floating base is widespread because they feature a higher flexibility regarding contact changes compared to dynamic models with a fixed base. In general, a central body within the kinematic structure of the robot is chosen as a base link, such as the hip or the trunk. Some works also utilize the center of mass (CoM) as a base, since it represents an essential quantity for balancing. Here, we will follow the proposition of [20] by defining a CoM frame \mathcal{C} , which is located at the CoM and has the same orientation of the hip. Let $\mathbf{x}_c \in \mathbb{R}^3$ and $\mathbf{R}_c \in \mathcal{SO}(3)$ denote the position and orientation of the frame \mathcal{C} with respect to the world frame \mathcal{W} . The corresponding translational and rotational velocities are $\dot{\mathbf{x}}_c$ and $\boldsymbol{\omega}_c$, respectively. Based on the n joint angles $\mathbf{q} \in \mathbb{R}^n$ and $\mathbf{v}_c = (\dot{\mathbf{x}}_c^T, \boldsymbol{\omega}_c^T)^T$, the dynamics of the humanoid robot is given by

$$\underbrace{\mathbf{M}}_{\boldsymbol{\nu}} \underbrace{\begin{pmatrix} \dot{\mathbf{v}}_c \\ \ddot{\mathbf{q}} \end{pmatrix}}_{\dot{\boldsymbol{\nu}}} + \underbrace{\mathbf{C}}_{\boldsymbol{\nu}} \underbrace{\begin{pmatrix} \mathbf{v}_c \\ \dot{\mathbf{q}} \end{pmatrix}}_{\boldsymbol{\nu}} + \underbrace{\begin{pmatrix} m\mathbf{g}_0 \\ \mathbf{0} \end{pmatrix}}_{\mathbf{g}} = \begin{pmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{pmatrix} + \boldsymbol{\tau}_{\text{ext}}. \quad (1)$$

Herein, $\mathbf{M} \in \mathbb{R}^{(6+n) \times (6+n)}$ and $\mathbf{C} \in \mathbb{R}^{(6+n) \times (6+n)}$ denote the inertia and Coriolis/centrifugal matrix, respectively. The gravitational torques are given by $\mathbf{g} \in \mathbb{R}^{6+n}$ with m denoting the overall mass of the robot and $\mathbf{g}_0 \in \mathbb{R}^6$ the gravitational acceleration¹. The joint torques are given by $\boldsymbol{\tau} \in \mathbb{R}^n$. The influence of external wrenches acting on the robot is taken into account by the generalized torque vector $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^{6+n}$.

Let us divide the Ψ end-effectors into two subgroups [20]: The first one is referred to as “balancing end-effector” (*bal*) and contains the ψ end-effectors that are used by the robot to support itself (usually the feet). The remaining end effectors are called “interaction end-effectors” (*int*), as they are still free to be used in a manipulation or interaction task (usually the hands). Based on this definition, the Cartesian velocities of the end effectors $\mathbf{v} \in \mathbb{R}^{6\Psi}$ are given by

$$\mathbf{v} = \begin{pmatrix} \mathbf{v}_{\text{bal}} \\ \mathbf{v}_{\text{int}} \end{pmatrix} = \begin{bmatrix} \mathbf{J}_{\text{bal}} \\ \mathbf{J}_{\text{int}} \end{bmatrix} \boldsymbol{\nu} = \mathbf{J} \boldsymbol{\nu}. \quad (2)$$

with the Jacobian matrix $\mathbf{J} \in \mathbb{R}^{6\Psi \times (6+n)}$ and $\mathbf{v}_{\text{bal}} \in \mathbb{R}^{6\psi}$, $\mathbf{v}_{\text{int}} \in \mathbb{R}^{6(\Psi-\psi)}$, $\mathbf{J}_{\text{bal}} \in \mathbb{R}^{6\psi \times (6+n)}$, $\mathbf{J}_{\text{int}} \in \mathbb{R}^{6(\Psi-\psi) \times (6+n)}$. In the case where all external disturbances act solely at the end-effectors, $\boldsymbol{\tau}_{\text{ext}}$ simplifies to

$$\boldsymbol{\tau}_{\text{ext}} = \mathbf{J}^T \mathbf{F}_{\text{ext}} \quad (3)$$

¹Note that \mathbf{g}_0 is six-dimensional by containing also the rotational DoFs.

B. Underlying Compliance Controller

This section gives a brief recapitulation of our balancing controller presented in [20]. The controller stabilizes the CoM by a Cartesian compliance, which applies a wrench $\mathbf{F}_c \in \mathbb{R}^6$ at the CoM frame \mathcal{C} . Each one of the interaction end-effectors is stabilized by another Cartesian compliance, with the resulting wrenches stacked into $\mathbf{F}_{\text{int}} \in \mathbb{R}^{6(\Psi-\psi)}$. In order to support the robot, the control algorithm computes a suitable set of balancing wrenches $\mathbf{F}_{\text{bal}} \in \mathbb{R}^{6\Psi}$ by solving the following quadratic optimization problem

$$\mathbf{F}_{\text{bal}}^{\text{opt}} = \underset{\mathbf{F}_{\text{bal}}}{\text{argmin}} \left(\mathbf{F}_{\text{bal}} - \mathbf{F}_{\text{bal}}^{\text{def}} \right)^T \mathbf{Q} \left(\mathbf{F}_{\text{bal}} - \mathbf{F}_{\text{bal}}^{\text{def}} \right) \quad (4)$$

with respect to

$$\mathbf{A}d_{\text{bal}}^T \mathbf{F}_{\text{bal}} + \mathbf{A}d_{\text{int}}^T \mathbf{F}_{\text{int}} = m\mathbf{g}_0 - \mathbf{F}_c \quad (5)$$

and

$$\begin{aligned} f_{k,z} &\geq f_{k,z}^{\min}, \\ \delta_k &\in \mathcal{S}_k, \\ |f_{k,x/y}| &\leq \tilde{\mu}_k f_{k,z}. \end{aligned} \quad (6)$$

The cost function (4) minimizes the deviation of \mathbf{F}_{bal} from a default wrench distribution $\mathbf{F}_{\text{bal}}^{\text{def}}$ considering the positive definite weighting matrix $\mathbf{Q} \in \mathbb{R}^{6\Psi \times 6\Psi}$. The default distribution $\mathbf{F}_{\text{bal}}^{\text{def}}$ is a tuning parameter, which can be set to half of the weight of the robot if the stance is symmetric. The equality constraint (5) represents the underactuation of the base by demanding that the influence of all commanded end-effector wrenches (\mathbf{F}_{bal} , \mathbf{F}_{int}) on the CoM must sum up to the compliance wrench \mathbf{F}_c plus gravity. For this, the Jacobian matrices $\mathbf{J}_{\text{bal}} = [\mathbf{A}d_{\text{bal}} \ \bar{\mathbf{J}}_{\text{bal}}]$ and $\mathbf{J}_{\text{int}} = [\mathbf{A}d_{\text{int}} \ \bar{\mathbf{J}}_{\text{int}}]$ are partitioned into $\mathbf{A}d_{\text{bal}} \in \mathbb{R}^{6\Psi \times 6}$, $\mathbf{A}d_{\text{int}} \in \mathbb{R}^{6(\Psi-\psi) \times 6}$, $\bar{\mathbf{J}}_{\text{bal}} \in \mathbb{R}^{6\Psi \times n}$, and $\bar{\mathbf{J}}_{\text{int}} \in \mathbb{R}^{6(\Psi-\psi) \times n}$. The first two are the stacked adjoint matrices of each end-effector, and relate a motion of the CoM frame \mathcal{C} with a motion of the end-effectors. The matrices $\bar{\mathbf{J}}_{\text{bal}}$ and $\bar{\mathbf{J}}_{\text{int}}$ describe the influence of a joint motion on the end-effectors. The inequality constraints (6) represent the contact model to which \mathbf{F}_{bal} is subjected to in order to account for unilaterality, the position of the Center of Pressure (CoP), and for friction of the balancing contacts. For each wrench within \mathbf{F}_{bal} , the force perpendicular to the contact surface \mathcal{S} is bounded from below by the minimum contact force $f_{k,z}^{\min}$ in order to prevent the end-effector from lifting off². Slippage is prevented by constraining the tangential forces $f_{k,x/y}$ to the friction cone given by $\tilde{\mu}_k$. The CoP δ_k is restricted to lie inside the contact surface \mathcal{S} to prevent the end-effector from tilting. From the contact model (6) and a given contact configuration, one can compute an equivalent support polygon, as shown in [24]. In order to achieve a static and stable balancing, the CoM projected to the ground floor must stay within the support polygon as well. This allows us to use the support polygon as a stability criterion in Sec. III-C.

²For the conducted experiments (see Sec. IV), $f_{k,z}^{\min}$ was set to 50 N in order to account for joint friction.

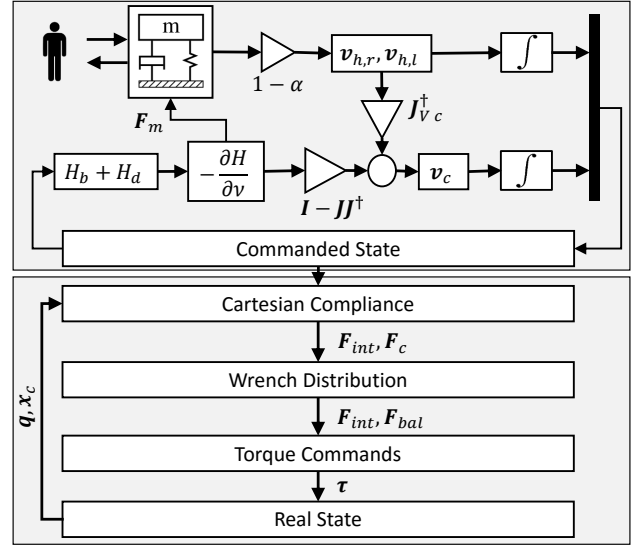


Fig. 2. A scheme of the proposed shared-control framework. The lower block represents the underlying compliance controller, while the upper block is the high-level teleoperation framework.

After computing a suitable wrench distribution $\mathbf{F}_{\text{bal}}^{\text{opt}}$, the end effector wrenches are mapped to joint space via

$$\boldsymbol{\tau} = - \begin{bmatrix} \bar{\mathbf{J}}_{\text{bal}}^T & \bar{\mathbf{J}}_{\text{int}}^T \end{bmatrix} \begin{pmatrix} \mathbf{F}_{\text{bal}}^{\text{opt}} \\ \mathbf{F}_{\text{int}} \end{pmatrix}. \quad (7)$$

III. SHARED-CONTROL ARCHITECTURE

A. Whole-Body Motion

While it is habitual to have the high-level ‘manipulation’ controller incorporated into the compliance controller itself, a different approach is opted for in this work. Indeed, the compliance controller described in the previous section is treated as a ‘packaged’ controller and topped with the needed higher level position/velocity whole-body motion controller, as shown in Fig. 2. This approach is remunerative as it allows the user to design a controller for a force/torque compliance-controlled robot from a pre-designed and tuned compliance controller without having to deal with its specificities.

Going back to Fig. 2, the shared-control teleoperation architecture is the upper gray zone. The user acts on an HMI sending velocity commands ($v_{m,r}$, $v_{m,l}$) through the right and left arms of the master device while receiving haptic cues ($\mathbf{F}_{m,r}$, $\mathbf{F}_{m,l}$) from potentials informing about the proximity to the edge of the support polygon and other physical constraints of the system. As the user gets closer to hitting these constraints, her/his commands are gradually scaled down to zero in order to ensure the stability of the system. On the other hand, the mentioned potentials are also used to generate a CoM motion v_c in the null-space of the motion of the hands, which are commanded by the user, and the feet (stable on the ground) in order to ensure that the system remains as far as possible from the constraints. The resulting velocities are integrated to generate new positions, which are then fed to the underlying compliance controller.

B. Master Side

As described before, and following the classical bilateral force-feedback teleoperation framework, we assume the presence of a master device consisting of a right and a left arm, through which the user sends velocity commands and receives force-feedback. The two master arms are modeled as generic (gravity pre-compensated) mechanical systems

$$\mathbf{M}_m(\mathbf{x}_m)\ddot{\mathbf{x}}_m + \mathbf{C}_m(\mathbf{x}_m, \dot{\mathbf{x}}_m)\dot{\mathbf{x}}_m = \begin{bmatrix} \mathbf{F}_{m,r} \\ \mathbf{F}_{m,l} \end{bmatrix} + \mathbf{F}_h \quad (8)$$

where $\mathbf{x}_m \in \mathbb{R}^{12}$ is the device cartesian configuration vector containing the pose of both the right and left master arms, $\mathbf{M}_m(\mathbf{x}_m) \in \mathbb{R}^{12 \times 12}$ is the positive-definite symmetric mass matrix, $\mathbf{C}_m(\mathbf{x}_m, \dot{\mathbf{x}}_m)\dot{\mathbf{x}}_m \in \mathbb{R}^{12 \times 12}$ accounts for Coriolis/centrifugal terms, $\mathbf{F}_h \in \mathbb{R}^{12}$ account for the forces applied by the human operator, and $\mathbf{F}_{m,r}, \mathbf{F}_{m,l} \in \mathbb{R}^6$ are the control forces on the right and the left arm, respectively. On the other hand, the individual linear and angular velocities of each of the master's right and left arms' end-effectors are denoted by $\mathbf{v}_{m,r} \in \mathbb{R}^6$ and $\mathbf{v}_{m,l} \in \mathbb{R}^6$ respectively.

The operator is given control over the hands of the robot through a direct Cartesian coupling. As the workspace of a human operator and the master arms is close to that of the humanoid, no scaling is implemented. However, the user can always “clutch” to move the master arms to a more convenient position without moving the slave. A velocity-velocity master-slave coupling is employed such that

$$\begin{cases} \mathbf{v}_{h,r} = (1 - \alpha)\tilde{\mathbf{R}}_k \mathbf{v}_{m,r} \\ \mathbf{v}_{h,l} = (1 - \alpha)\tilde{\mathbf{R}}_k \mathbf{v}_{m,l} \end{cases} \quad (9)$$

where $\mathbf{v}_{h,r}$ and $\mathbf{v}_{h,l} \in \mathbb{R}^6$ are the desired velocities of the right and left hands of the slave respectively, and $\tilde{\mathbf{R}}_k$ is the rotation map between the master and the slave velocities. $\alpha : 0 \mapsto 1$ is an activation function that goes from 0 to 1 as the system approaches any of the different constraints, with 0 being away from the constraint and 1 being exactly at it.

C. Balancing and Physical Constraints

In addition to executing the operator's commands, it is crucial to maintain the stability of the humanoid by ensuring its CoM remains within the support polygon³, as shown in Fig. 3. To this end, we define a cost function $H_b : \mathbb{R}^n \mapsto \mathbb{R}$ as a measure of the ‘balance’ of the humanoid, which goes to infinity as the user approaches the edges of the support polygon such that

$$H_b(\mathbf{p}_c) = \sum_i h_{i,b}(\mathbf{p}_c), \quad (10)$$

where $\mathbf{p}_c \in \mathbb{R}^3$ is the position of the center of mass, and $h_{i,b}$ is the cost function attached to each edge i of the support polygon and defined by

³Note that this condition is only valid for evaluating static stability. In the dynamic case, a condition could be used based e.g. on the capture point [25].

$$h_{i,b}(\mathbf{p}_c) = \begin{cases} \mu \tan\left(\frac{\pi}{2} \left(1 - \frac{d_i}{d_{l,i}}\right)\right)^2 & d_i < d_{l,i}, \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

with μ being a regulation gain ($\mu = 1$ in this work), d_i is the horizontal distance from the center of mass to each edge i of the support polygon (which is known from the fixed posture of the feet), and $d_{l,i}$ is the distance from the center of the support polygon to its i -th edge.

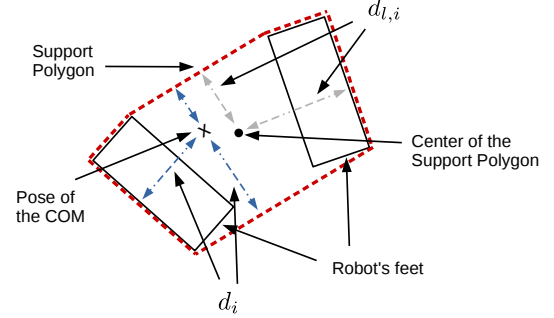


Fig. 3. An illustration of the support polygon for a robot standing on horizontal ground.

The time derivative of H_b can then be defined by

$$\dot{H}_b = \frac{\partial H_b}{\partial \mathbf{p}_c} \dot{\mathbf{p}}_c = \frac{\partial H_b}{\partial \mathbf{p}_c} \mathbf{J}_{\mathbf{p}_c} \boldsymbol{\nu}. \quad (12)$$

where $\mathbf{J}_{\mathbf{p}_c} = [\mathbf{I} \ \mathbf{0}] \in \mathbb{R}^{3 \times 6+n}$ is the jacobian mapping the state velocities $\boldsymbol{\nu}$ to $\dot{\mathbf{p}}_c$.

While the physical constraints of a humanoid robot range from workspace limits to joint limits, singularities and torque constraints, a representative workspace constraint was considered in this work consisting of the distance between the hands of the robot and its center of mass. Constraining this distance ensures that the robot maintains a “safe” posture and avoids a “full stretch” configuration that could push the robot to singularities and joint limits. While this constraint does not cover the various limitations of a humanoid, it is meant as a proof of concept and a representative constraint. The exploration of further constraints is a future work.

The constraint hereby considered is then to maintain the distance between each of the hands of the humanoid and its CoM within a predefined threshold d_{th} . The corresponding cost function H_d accounting for this physical constraint is defined as

$$H_d(\mathbf{p}_c, \mathbf{p}_{h,r}, \mathbf{p}_{h,l}) = h_{c,r}(\mathbf{p}_c, \mathbf{p}_{h,r}) + h_{c,l}(\mathbf{p}_c, \mathbf{p}_{h,l}) \quad (13)$$

where the cost function of each hand $h_{c,x}(\mathbf{p}_c, \mathbf{p}_{h,x})$ is

$$h_{c,x} = \begin{cases} \rho \tan\left(\frac{\pi}{2} \left(1 - \frac{d_{h,x} - d_s}{d_{th} - d_s}\right)\right)^2 & d_s < d_i < d_l, \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

$d_{h,x}$ in the above equation is the distance from each hand x to the center of mass, while d_s is the minimum distance after

which the potential starts, and ρ is a positive gain ($\rho = 1$ in this work). The derivative of the potential H_d is then

$$\begin{aligned}\dot{H}_d &= \begin{bmatrix} \frac{\partial H_d}{\partial \mathbf{p}_c} & \frac{\partial H_d}{\partial \mathbf{p}_{h,r}} & \frac{\partial H_d}{\partial \mathbf{p}_{h,l}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{p}}_c \\ \dot{\mathbf{p}}_{h,r} \\ \dot{\mathbf{p}}_{h,l} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial H_d}{\partial \mathbf{p}_c} & \frac{\partial H_d}{\partial \mathbf{p}_{h,r}} & \frac{\partial H_d}{\partial \mathbf{p}_{h,l}} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\mathbf{p}_c} \\ \mathbf{J}_{\mathbf{p}_{h,r}} \\ \mathbf{J}_{\mathbf{p}_{h,l}} \end{bmatrix} \boldsymbol{\nu},\end{aligned}\quad (15)$$

where $\mathbf{J}_{\mathbf{p}_{h,r}}$ and $\mathbf{J}_{\mathbf{p}_{h,l}}$ are the jacobians linking the linear velocities of the right and left hand to the state velocity $\boldsymbol{\nu}$, respectively.

The desired state velocity $\boldsymbol{\nu}_{des}$ to minimize the potentials and make sure the system constraints are respected can then be defined as the negative of the transpose of the gradient of potentials H_b and H_d (eq. (12) and eq. (15)) such that

$$\boldsymbol{\nu}_{des} = - \left(\begin{bmatrix} \frac{\partial H_d}{\partial \mathbf{p}_c} & \frac{\partial H_d}{\partial \mathbf{p}_{h,r}} & \frac{\partial H_d}{\partial \mathbf{p}_{h,l}} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\mathbf{p}_c} \\ \mathbf{J}_{\mathbf{p}_{h,r}} \\ \mathbf{J}_{\mathbf{p}_{h,l}} \end{bmatrix} \right)^T - \left(\frac{\partial H_b}{\partial \mathbf{p}_c} \mathbf{J}_{\mathbf{p}_c} \right)^T \quad (16)$$

D. CoM Control Law

The motion of the center of mass is designed to follow the commands of the human operator while fixing the feet on the ground as a primary task, and maintaining the balance of the robot in their null-space. We commence now by explaining the autonomous balancing algorithm governing this behavior.

The robot's state velocity vector is mapped to the velocities of the hands and the feet (which are stationary in the case at hand) through a jacobian \mathbf{J}

$$\begin{bmatrix} \mathbf{v}_{h,r} \\ \mathbf{v}_{h,l} \\ \mathbf{v}_{f,r} \\ \mathbf{v}_{f,l} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \mathbf{v}_c \\ \dot{\mathbf{q}} \end{bmatrix}. \quad (17)$$

The impact of the velocity inputs from the human operator on the velocity of the CoM, dubbed as its primary task, can then be retrieved by inverting the previous equation such that

$$\mathbf{v}_c = \mathbf{A} \mathbf{d}^\dagger \begin{bmatrix} \mathbf{v}_{h,r} \\ \mathbf{v}_{h,l} \\ \mathbf{v}_{f,r} \\ \mathbf{v}_{f,l} \end{bmatrix} \quad (18)$$

with $\mathbf{A} \mathbf{d}^\dagger$ being the Moore pseudo-inverse of $\mathbf{A} \mathbf{d} = [\mathbf{A} \mathbf{d}_{bal}^T \mathbf{A} \mathbf{d}_{int}^T]^T$, which is the upper part of Jacobian \mathbf{J} corresponding to the velocity of the center of mass \mathbf{v}_c . The null-space balancing motion is then added to (18), and the resulting full control law becomes

$$\mathbf{v}_c = \mathbf{A} \mathbf{d}^\dagger \begin{bmatrix} \mathbf{v}_{h,r} \\ \mathbf{v}_{h,l} \\ \mathbf{v}_{f,r} \\ \mathbf{v}_{f,l} \end{bmatrix} + \mathbf{N} \boldsymbol{\nu}_{des} \quad (19)$$

where $\mathbf{N} = (\mathbf{I} - \mathbf{J} \mathbf{J}^\dagger)$ is the null-space projector.

E. Haptic Feedback

In case of proximity to a constraint, force-feedback is given to the operator to guide her/him away from the undesired configuration. Moreover, if the operator keeps pushing towards the constraint, her/his velocity commands are tuned down to prevent the system from hitting the constraint.

A proximity function $\alpha = \max(\alpha_i(d_i))$ is employed for that purpose, where $\alpha_i : 0 \mapsto 1$ is the proximity function associated to each constraint i . The constraints in this case include the different edges of the support polygon and the distances from the hands to the CoM. $\alpha_i(d_i)$ increases linearly from 0 to 1 as a function of the distance to the corresponding constraint d_i , such that it is 0 before a predefined threshold and reaches 1 at the constraint

$$\begin{cases} \alpha_i(d_i) : 0 \mapsto 1 \\ d_i : d_{p,i} \mapsto d_{l,i}. \end{cases} \quad (20)$$

with $d_{p,i}$ being a maximum threshold after which α_i becomes active, and $d_{l,i}$ is the limit of the constraint.

On the other hand, the desired motion direction of the hands $[\tilde{\mathbf{v}}_{h,r}, \tilde{\mathbf{v}}_{h,l}]$ that ensures the minimization of the potentials H_b and H_d can be retrieved as

$$\begin{bmatrix} \tilde{\mathbf{v}}_{h,r} \\ \tilde{\mathbf{v}}_{h,l} \end{bmatrix} = - \begin{bmatrix} \mathbf{J}_{h,r} \\ \mathbf{J}_{h,l} \end{bmatrix} \mathbf{N}_{bal} \boldsymbol{\nu}_{des}, \quad (21)$$

where $\mathbf{N}_{bal} = (\mathbf{I} - \mathbf{J}_{bal} \mathbf{J}_{bal}^\dagger)$ is the null-space projector ensuring that the generated motion does not impact the position of the feet. \mathbf{J}_{bal} is the jacobian mapping the state velocity $\boldsymbol{\nu}$ to the velocity of the feet $[\mathbf{v}_{f,r}, \mathbf{v}_{f,l}]$.

The input velocities of the human operator are then tuned down if the robot is in the proximity of a constraint, i.e. $\alpha > 0$, and the direction commanded by the operator is opposing the desired motion directions $[\tilde{\mathbf{v}}_{h,r}, \tilde{\mathbf{v}}_{h,l}]$

$$\begin{cases} \mathbf{v}_{h,r} = (1 - \alpha) \tilde{\mathbf{R}}_k^T \tilde{\mathbf{v}}_{h,r} & \text{if } \tilde{\mathbf{v}}_{h,r}^T \tilde{\mathbf{R}}_k \mathbf{v}_{m,r} < 0 \\ \mathbf{v}_{h,l} = (1 - \alpha) \tilde{\mathbf{R}}_k^T \tilde{\mathbf{v}}_{h,l} & \text{if } \tilde{\mathbf{v}}_{h,l}^T \tilde{\mathbf{R}}_k \mathbf{v}_{m,l} < 0 \end{cases} \quad (22)$$

The forces received by the operator on the right and left arms are defined by

$$\begin{cases} \mathbf{F}_{m,r} = F_{max} \alpha \tilde{\mathbf{v}}_{h,r} / \|\tilde{\mathbf{v}}_{h,r}\| + \mathbf{B} \mathbf{v}_{m,r} \\ \mathbf{F}_{m,l} = F_{max} \alpha \tilde{\mathbf{v}}_{h,l} / \|\tilde{\mathbf{v}}_{h,l}\| + \mathbf{B} \mathbf{v}_{m,l} \end{cases} \quad (23)$$

where F_{max} is a design parameter defining the maximum force expected from the operator, and $\mathbf{B} \in \mathbb{R}^{6 \times 6}$ is a damping matrix.

This design of the haptic interface is unique in that it is not a pure resistive force stopping the user as she/he approaches a constraint or another system limit. On the contrary, the described force cues are an active guidance providing the user with several solutions for avoiding the system constraints over the 12 DoF she/he is commanding.

Note that the behavior of the force feedback in eq. (23) is akin to a spring-damper system, as α is directly proportional to the distance between the CoM and the support polygon

boundaries. The potentials H_b and H_d do go to infinity in the proximity of the edges of the support polygon, but the haptic feedback given to the operator adopts the direction of the gradient of these potentials (but not its magnitude). While this is not needed for passivity, as the potential can be proven passive in itself, it allows for a better design of the haptic interface, since any haptic device has limited capabilities for force generation and the design of the haptic interface must account for these limitations.

IV. EXPERIMENTS AND RESULTS

This section describes the experiments conducted to validate the described approach, and discusses the obtained results.

Our experimental setup uses HUG on the master side. HUG is a haptic user interface system composed by two light-weight robotic arms with a workspace of 1.1 m and a nominal payload of 20 kg each [26]. On the slave side, we use TORO, a humanoid robot developed at DLR [27]. It has a height of 1.74 m, a weight of 76.4 kg, and 25 joints in total (not counting the neck and the hands). In the presented experiments, the feet of TORO are in contact with the ground floor in order to support the robot (balancing end-effectors). The hands (interaction end-effectors) were directly commanded by the operator via HUG. The joints in the neck and hands were not used. Due to the feet contacts ($2 \times 6 = 12$ DoF) and the user input for the hands ($2 \times 6 = 12$ DoF), the robot can only move 7 DoF in order to maintain balance (6 DoF attached to the floating-base (frame \mathcal{C}) plus 1 remaining DoF).

In order to test the behavior of the different parts of the system, we distinguish two major components: the null-space autonomous balancer acting on the CoM to keep the system as far as possible from constraints, and the haptic guidance informing the user of the proximity to any of these constraints and the possible directions to avoid them. Three experiments were performed while activating or deactivating these two components, to assess their impact on the general behaviour of the system. The video attached to this paper provides further insight on the performed experiments.

A. Experiment I

In the first experiment, the haptic guidance and the null-space balancer were both deactivated, and the user was given unrestricted control over the hands of the humanoid.

The user was asked to reach as far as possible with the hands of the humanoid along the x-axis (which is pointing forward). Fig. 4 reports on the obtained results. Fig. 4a and Fig. 4b show the x-position of the hands and the CoM respectively throughout the experiment, while Fig. 4c shows the variation of the potentials H_b and H_d . Note that this is just a visualization of the potentials, as they were not active in this experiment. On the other hand, in Fig. 4b the horizontal dotted line represents the edge of the support polygon along the positive x-direction. The center of the support polygon was at 1 cm from the world frame in this scenario, with its edges at -3 cm and 7 cm along the x-axis.

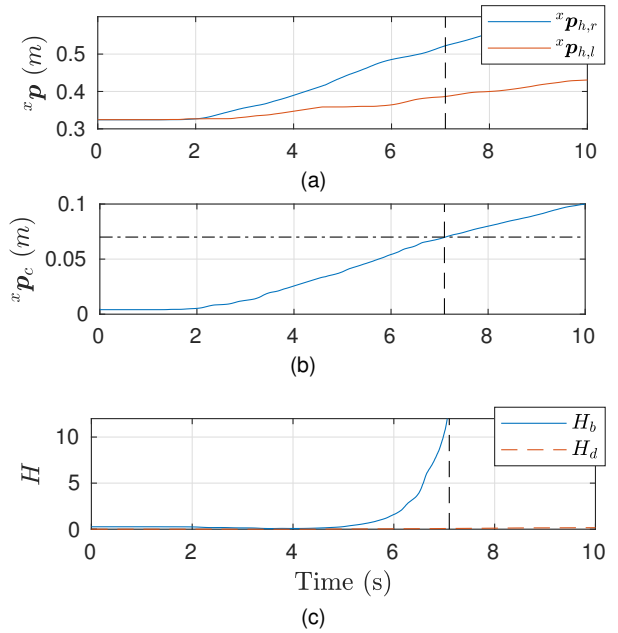


Fig. 4. Experiment I: No haptic guidance is provided, and the null space balancer is deactivated. (a) and (b) show the position of the hands and CoM along the x-axis, respectively, while (c) shows the system potentials H_b and H_d .

Observing the figures, the maximum reach of the hands of the humanoid before hitting the support polygon's edge was 52.7 cm for the right arm (the vertical dotted line denotes the moment at which the CoM crossed the edge of the support polygon). In fact, the humanoid then started to tip over, losing contact on its right foot, and the experiment was called to a halt. Note that the potential H_b increased significantly as the edge of the support polygon was closer.

The results of this experiment are also depicted in Fig. 7, where Fig. 7a shows the trajectories followed by the right hand (in blue) and the CoM (in red), and Fig. 7d depicts the final posture of the robot after the right foot lost grip.

B. Experiment II

Following on the previous experiment, haptic guidance was activated along with the restriction applied on the operator's commands when approaching a constraint (by exploiting the proximity measure α , check eq. (22)). The null-space autonomous balancer was, however, still not active. The same experiment was repeated and the user was asked again to reach as far as possible. The results are reported in Fig. 5. Fig. 5b plots the forces fed to the operator (along the x-axis) on the end-effectors of the master device. On the other hand, Fig. 5a and Fig. 5c show the x-position of the slave hands (right and left) and the center of mass, respectively, as the experiment proceeds. Finally Fig. 5d depicts the two potential functions describing the stability of the robot and its physical constraints.

An interesting behavior is observed in this experiment. As the user moved the hands of the robot forward, the CoM started approaching the edge of the support polygon, and this was reflected as an increase in the cost functions and an active force on both hands along the negative x direction

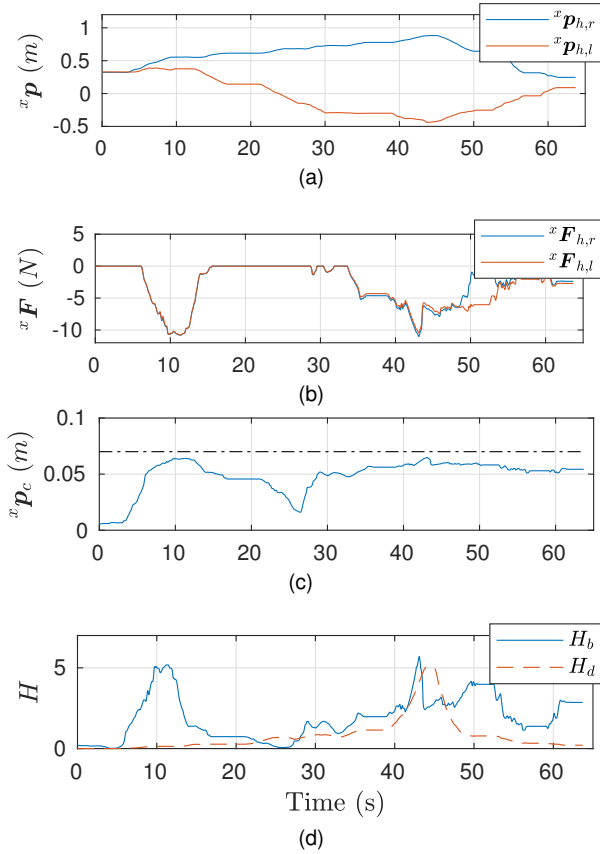


Fig. 5. Experiment II: Haptic guidance is provided to the operator, but the null-space balancer is deactivated. (b) shows the force cues fed to the operator along the x-axis on the right and left hand, respectively, while (a) and (c) report on the x-position of the hands and the CoM. (d) plots the potentials H_b and H_d .

(see $t=10$ sec). The user then reactively started moving the left hand backwards, following the haptic guidance. This decreased the cost function, allowing him to further push the right hand forward. In fact, this strategy, inspired by the informative haptic guidance, helped the user to reach 34.9 cm farther than the first experiment ($t=45$ sec), without impacting the stability of the robot.

The followed trajectories and final posture of the robot for experiment II are reported in Fig. 7b and Fig. 7e.

C. Experiment III

The null-space autonomous balancer was activated for the third experiment, in addition to having the haptic guidance active as well. The experiment followed the same procedure as before, and the results are reported in Fig. 6.

An interesting figure to observe in this experiment is Fig. 6c, which shows the evolution of the x-position of the CoM. In contrast to the previous two experiments, the CoM remained close to the center of the support polygon for a significant portion of the experiment. In fact, both the right and left hands were close to a 60 cm reach before the CoM started moving forward. It was then pushed forward as the hands were being moved farther ahead, thus driving the robot to be more stretched and increasing the potential H_d , as can be observed in Fig. 6d. The resulting maximum reach was 91.4 cm, a bit higher than in experiment II. However, the

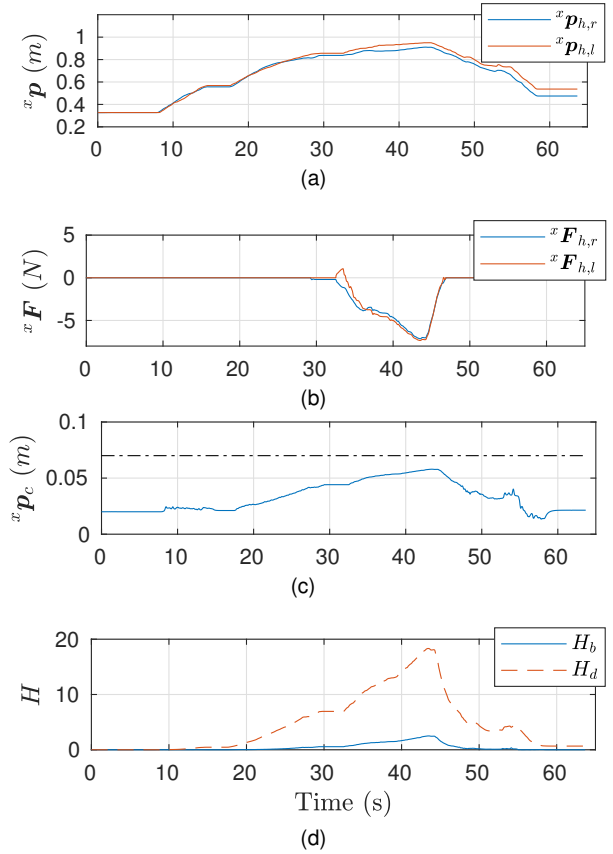


Fig. 6. Experiment III: Haptic guidance is provided to the operator, and the null-space balancer is active. (b) shows the force cues fed to the operator along the x-axis on the right and left hand, respectively, while (a) and (c) show the x-position of the hands and the CoM. (d) depicts the potentials H_b and H_d .

user was able to push both hands forward instead of one, thus allowing for more manipulation options.

On the other hand, Fig. 6b shows that the forces received throughout this experiment were negligible, except for $t=[33,48]$ s when the hands were stretched at more than 85 cm. This is important in that autonomy, while accounting for all the operator's commands, was able to successfully command the null-space and provide the operator with the maximum workspace possible with minimal disturbance.

The followed trajectories and final posture of the robot in this experiment are reported in Fig. 7c, and Fig. 7f.

V. CONCLUSIONS

This paper presented a shared-control architecture for tele-operating a torque-controlled humanoid robot. An operator was given control over the hands of the humanoid, while the whole-body motion was governed by a null-space balancer acting in the null-space of the operator's commands. A novel approach for providing haptic feedback is introduced, where the user is fed with high-level informative haptic cues informing her/him about the impact of her/his potential actions on the robot's balance. This approach bridges the gap between the task itself and the different constraints of the system, thus allowing the operator to adapt her/his approach for a successful task execution within the constraints of the system.

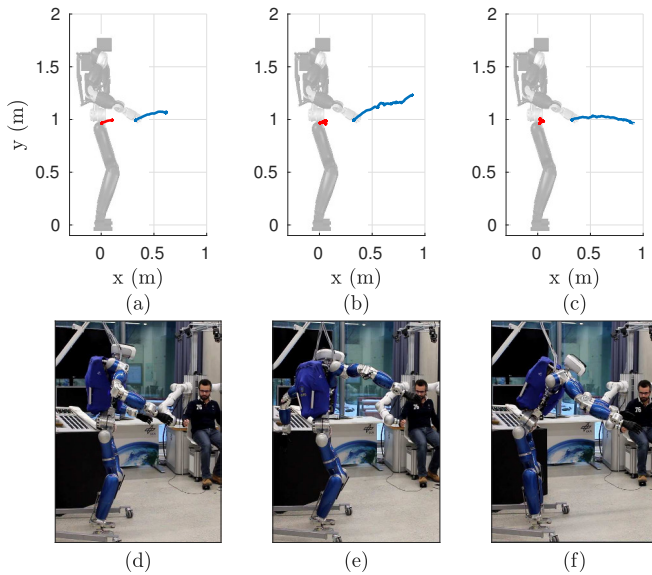


Fig. 7. Comparison of the conducted experiments: The top row shows the trajectory of the right hand and the CoM in the sagittal plane of the robot. The bottom row shows the final pose of the robot with the largest achievable reach of the hands. (a) and (d) report on experiment I, (b) and (e) on experiment II, and (c) and (f) on experiment III

On the other hand, the authors are looking forward to explore different constraints of the system both in Cartesian space (collision avoidance) and in joint space (joint limits, singularities, ...), and study the possibility of incorporating them in the architecture in the future. Another potential future work is the consideration of variable support polygons, which change depending on the (changing) posture/configuration of the robot. Finally, incorporating dynamics into the approach to provide the operator with a meaningful instantaneous guidance during a dynamic behavior, like walking, is another future challenge.

ACKNOWLEDGMENT

This work has received funding from the European Union Horizon 2020 Research and Innovation programme under grant agreement No. 645582 (RoMaNS), and H2020-ICT-645097 (COMANOID).

REFERENCES

- [1] E. S. Neo, K. Yokoi, S. Kajita, and K. Tanie, "Whole-body motion generation integrating operator's intention and robot's autonomy in controlling humanoid robots," *IEEE Trans. Robotics*, vol. 23, no. 4, pp. 763–775, 2007.
- [2] K. Hertkorn, B. Weber, P. Kremer, M. A. Roa, and C. Borst, "Assistance for telepresence using online grasp planning," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2013, pp. 507–513.
- [3] A. Ghalamzan, F. Abi-Farraj, P. Robuffo Giordano, and R. Stolkin, "Human-in-the-loop optimisation: mixed initiative grasping for optimally facilitating post-grasp manipulative actions," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2017, pp. 3386–3393.
- [4] F. Abi-Farraj, T. Osa, N. Pedemonte, J. Peters, G. Neumann, and P. Robuffo Giordano, "A learning-based shared control architecture for interactive task execution," in *IEEE Int. Conf. Robotics and Automation*, 2017, pp. 329–335.
- [5] N. Pedemonte, F. Abi-Farraj, and P. Robuffo Giordano, "Visual-based shared control for remote telemanipulation with integral haptic feedback," in *IEEE Int. Conf. Robotics and Automation*, 2017, pp. 5342–5349.
- [6] J. J. Abbott, P. Marayong, and A. M. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Robotics research*. Springer, 2007, pp. 49–64.
- [7] F. Abi-Farraj, N. Pedemonte, and P. Robuffo Giordano, "A visual-based shared control architecture for remote telemanipulation," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2016, pp. 4266–4273.
- [8] R. O. Ambrose, H. Aldridge, R. S. Askew, R. R. Burrage, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark, "Robonaut: NASA's space humanoid," *IEEE Intelligent Systems and their Applications*, vol. 15, no. 4, pp. 57–63, 2000.
- [9] J. Iqbal, A. M. Tahir, R. ul Islam, and R. un Nabi, "Robotics for nuclear power plants - challenges and future perspectives," in *Int. Conf. Applied Robotics for the Power Industry*, 2012, pp. 151–156.
- [10] H. Hasunuma, M. Kobayashi, H. Moriyama, T. Itoko, Y. Yanagihara, T. Ueno, K. Ohya, and K. Yokoi, "A tele-operated humanoid robot drives a lift truck," in *IEEE Int. Conf. Robotics and Automation*, vol. 3, 2002, pp. 2246–2252.
- [11] H. Song, H. Shin, H. You, J. Hong, and D. H. Shim, "Toward autonomous aircraft piloting by a humanoid robot: Hardware and control algorithm design," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2016, pp. 398–403.
- [12] S. Tachi, K. Komoriya, K. Sawada, T. Nishiyama, T. Itoko, M. Kobayashi, and K. Inoue, "Teleexistence cockpit for humanoid robot control," *Advanced Robotics*, vol. 17, no. 3, pp. 199–217, 2003.
- [13] K. Yokoi, K. Nakashima, M. Kobayashi, H. Mihune, H. Hasunuma, Y. Yanagihara, T. Ueno, T. Gokuyuu, and K. Endou, "A tele-operated humanoid operator," *Int. J. Robotics Research*, vol. 25, no. 5-6, pp. 593–602, 2006.
- [14] M. Zucker, S. Joo, M. X. Grey, C. Rasmussen, E. Huang, M. Stilman, and A. Bobick, "A general-purpose system for teleoperation of the DRC-HUBO humanoid robot," *J. Field Robotics*, vol. 32, no. 3, pp. 336–351, 2015.
- [15] E. S. Neo, K. Yokoi, S. Kajita, and K. Tanie, "Whole-body motion generation integrating operator's intention and robot's autonomy in controlling humanoid robots," *IEEE Trans. on Robotics*, vol. 23, no. 4, pp. 763–775, 2007.
- [16] E. G. Estrelinha, *Tele-operation of a humanoid robot using haptics and load sensors*. Ph.D. dissertation, Universidade de Aveiro, 2013.
- [17] J. Oliveira, V. Ferreira, and F. Teixeira, "Bimanual haptics for humanoid robot teleoperation using ROS and V-REP," in *IEEE Int. Conf. Autonomous Robot Systems and Competitions*, 2015, pp. 174–179.
- [18] A. Brygo, I. Sarakoglou, N. Tsagarakis, and D. G. Caldwell, "Tele-manipulation with a humanoid robot under autonomous joint impedance regulation and vibrotactile balancing feedback," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 862–867.
- [19] J. Ramos, A. Wang, and S. Kim, "A balance feedback human machine interface for humanoid teleoperation in dynamic tasks," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2015, pp. 4229–4235.
- [20] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *Int. J. Robotics Research*, vol. 35, no. 12, pp. 1522 – 1543, 2016.
- [21] M. Mistry, J. Buchli, and S. Schaal, "Inverse dynamics control of floating base systems using orthogonal decomposition," in *IEEE Int. Conf. Robotics and Automation*, 2010, pp. 3406 – 3412.
- [22] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, "Optimal distribution of contact forces with inverse dynamics control," *Int. J. Robotics Research*, vol. 32, no. 3, pp. 280 – 298, 2013.
- [23] L. Sentis, "Compliant control of whole-body multi-contact behaviors in humanoid robots," in *Motion Planning for Humanoid Robots*, K. Harada, E. Yoshida, and K. Yokoi, Eds. Springer, 2010, pp. 29–66.
- [24] S. Caron and A. Kheddar, "Multi-contact walking pattern generation based on model preview control of 3D COM accelerations," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 550 – 557.
- [25] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 200 – 207.
- [26] M. Sagardia, T. Hulin, K. Hertkorn, P. Kremer, and S. Schätzle, "A platform for bimanual virtual assembly training with haptic feedback in large multi-object environments," in *ACM Conf. Virtual Reality Software and Technology*, 2016, pp. 153–162.
- [27] J. Engelsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schffer, "Overview of the torque-controlled humanoid robot TORO," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 916–923.